

## Fig. 2A

BWRQ packet

(write request for data block packet)

transmitted first			٠.	. 0	
destination_ID	tl	rt	tcode	pri	
source_ID	*		*		
destinati	on_offse	t			
data_length extended_tco			d_tcode	e 	
heade	r_CRC	11		1 1 1	
data_	field				
zero pad bytes(if necessary)					
data	_CRC				

transmitted last

Fig. 2B

QWRQ packet

(write request for data quadlet packet) transmitted first

31				0		
destination_ID	tl	rt	tcode	pri		
source_ID						
destination_offset						
quadle	t_data		1 ' + _4 _ 1			
heade	r_CRC					

transmitted last

## Fig. 2C

WRS packet

(write response packet)

ansmi	tted first					0	
d	estination_ID	1	tl	rt	tcode	pri	
	source_ID	rcod	le	r	reserved		
	r	eserved	1 1				
	he	ader_CR	<u> </u>				
	<del></del>		C				

transmitted last

Fig. 2D

BRRQ packet (read request for data block packet)

transmitted first				0
destination_ID	tl	rt	tcode	pri
source_ID			•	
destinati	on_offset		<u> </u>	
data_length	ext	ende	d_tcode	<b>)</b>
heade	r_CRC			

transmitted last

## Fig. 2E

BRRS packet (read response for data block packet)

transmitted first				0		
destination_ID	tl rt tcode 1					
source_ID						
destinati	on_offset		L			
data_length	extended_tcode					
header_CRC						
data_field						
zero padding(if necessary)						
data_CRC						

transmitted last

Fig. 3A

#### internal BWRQ packet

spd	tl	rt	tcode	pri		
destination_ID			18			
destinati	on_offset	<b> </b>	L. 1 . 1 . 1			
data_length	extended_tcode					
block data						
zero padding(if necessary)						

Fig. 3B

#### internal QWRQ packet

spd	tl	rt	tcode	pri			
destination_ID							
destination_offset							
quadlet_data							

Fig. 3C

#### internal WRS packet

destination_ID	tl	rt	tcode	pri
source_ID	rcode			
spd				ackSent

Fig. 3D

### internal BRRQ packet

spd	tl	rt	tcode	pri		
destination_ID						
destination_offset						
data_length						

# Fig. 3E

#### internal BRRS packet

destination_ID	tl	rt	tcode	pri	
source_ID			· .		
destinati	on_offset			<del>-</del> -	
data_length	extended_tcode				
block data					
zero pa	dding(if	nec	essary)		
spd				ackSent	

Fig. 4

target 2 (optical disk drive)	ORB_POINTER	analyze ORB		<u>^</u>	^	<u></u>
	Transfer ORB's address at initiator (Issue BWRQ to ORB_POINTER register at target) (Send WRS responsive to the BWRQ)	Request transmission of the ORB stored at the address (Issue BRRQ to the ORB's address at initiator) Send the ORB to target (Issue BRRS with the ORB stored in data field)	Store data at the address specified by the ORB (Issue BWRQ)	(Send ACK representing ack_complete or WRS representing resp_complete)  Store data at the address specified by the ORB  (Issue BWRQ)	(Send ACK representing ack_complete or WRS representing resp_complete)	Send status for the ORB  (Issue BWRQ to status_FIFO specified by the ORB)  (Send WRS representing resp_complete responsive to the BWRQ)
initiator 1 (host PC) create ORB	(I) <	(2)		(3)		(4)

- inform reception of packet addressed to CPU BWRQ #3 → time Ack Comp response detection WRS WRS Ack Pend wait for response BWRQ #3 data #3 BRRQ BWRQ #3 BRROZ BRRQ suspension -Ack Pend data #3 BWRQ #2 header #3 Ack Comp response \_ detection WRS WRS wait for response data #2 BWRQ #2 BWRQ #2 data #2 Ack Pend header #2 BWRQ Lboot data exchange data #1 data #1 BWRQ #1 BWRQ #1 header #1 က DMAC 13 10  $\infty$ 6 1394 bus 7 П വ 9 RFIL DTRF PPRC TFIL ATF ARF CPU

Fig. 6A









